**package** principal;

**import** java.util.Locale;

**import** java.util.Scanner;

**public** **class** Program {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

// uma constante é quando o seu valor não muda

**final** **double** PI = 3.1415;

// realiza a potencia o primeiro numero é a base e o segundo é a potencia

System.***out***.println(Math.*pow*(2, 2));

// passagem de 2 numeros como parametro retorna o maior deles

System.***out***.println(Math.*max*(5, 20));

// é o inverso do max, ou seja, retorna o mínimo na passagem de 2 valores

System.***out***.println(Math.*min*(5, 20));

//retorna a raiz quadrada de um numero

System.***out***.println(Math.*sqrt*(81));

//arredonda o numero (para mais ou para menos)

System.***out***.println(Math.*round*(25.6578));

//podemos realizar calculos com valores retornados pelo calculo de Math.

System.***out***.println(2 + Math.*sqrt*(18));

Scanner leia = **new** Scanner (System.***in***);

Locale ponto = **new** Locale ("en", "us");

//System.out.println("Entre com o valor do dolar: ");

//double dolar = leia.useLocale(ponto).nextDouble();

//System.out.println("Valor dolar digitado: " + dolar);

String nome = "Fernanda Guedes";

**char** sexo = 'F';

**int** idade = 32;

**double** altura = 1.588888414;

//o printf serve para formatar, ou seja, quando eu coloco %s leia uma String, %c leia uma char,

//%d leia um inteiro, % f leia um double. %.2f mostra apenas duas casas decimais no double e arredonda o valor.

System.***out***.printf("O meu nome é %s, sou do sexo %c, tenho %d anos, minha altura é: %.2f", nome, sexo, idade, altura);

/\*

\* operadores de comparação

\* > maior que

\* < menor que

\* >= maior ou igual a

\* <= menor ou igual a

\* != diferente de

\* == igual a

\*/

/\* operadores lógicos

\* && = e

\* || = ou

\* ! = negação

\*/

A tabela abaixo exemplifica os valores lógicos de (e) and (ou)

leia.close();

}

}

OPERADOR LÓGICO && (E) AMBAS PRECISAM SER VERDADEIRA PARA DEVOLVER VERDADEIRA

|  |  |  |
| --- | --- | --- |
| VALOR | VALOR | DEVOLVE |
| V | F | F |
| F | V | F |
| V | V | V |
| F | F | F |

OPERADOR LÓGICO || (OU) PRECISA APENAS DE UM VALOR VERDADEIRO E JÁ DEVOLVE VERDADEIRO, PARA DEVOLVER FALSO, AMBAS PRECISAM SER FALSAS.

|  |  |  |
| --- | --- | --- |
| VALOR | VALOR | DEVOLVE |
| V | F | V |
| F | V | V |
| V | V | V |
| F | F | F |

**int** numero1 = 10;

**int** numero2 = 25;

**boolean** bool = **false**;

/\*

\* o if significa "se", e a chave no final delimita o meu if, ou seja, o meu

\* comando if terminou. o else significa "se não" o boolean retorna "verdadeiro"

\* ou "falso" quando eu nego a minha variável ela muda o valor (como na tabela

\* verdade) ex: a variavel bool = false para negar a variavel usa : !bool = true

\* ! = negação da variável

\*/

**if** (numero1 > numero2) {

System.***out***.println("número 1 é o maior");

} **else** {

System.***out***.println("número 2 é o maior");

}

**if** (bool == **false**) {

System.***out***.println("Verdadeiro");

} **else** {

System.***out***.println("Falso");

}

// Exemplo usando resto da divisão para saber se é impar ou par

Scanner leia = **new** Scanner(System.***in***);

**int** numero;

System.***out***.println("Entre com um número para saber se é par ou ímpar");

numero = leia.nextInt();

leia.close();

**if** (numero % 2 == 0) {

System.***out***.println("número par!");

} **else** {

System.***out***.println("número ímpar!");

}

}

}